ITEA Journal 2010; 31: 499–511

Joint Sensor: Security Test and Evaluation Embedded in a Production Network Sensor Cloud

Tim Owen, Rob Scott, and Roy Campbell, Ph.D.
Defense Research and Engineering Network,
High Performance Computing Modernization Program, Lorton, Virginia

A great security posture inherently requires that cyber operations employ the latest discoveries in emerging security research to keep in step with trends in attack methodologies. The most trenchant cyber security research to date employs actual network data to ensure sensing algorithms and defense methodologies are effective in real-world scenarios. This approach often requires discernments to be made as temporally close to the observed events as possible to allow rapid adaptability of the security posture upon detection of an anomaly. Traditional security architectures, on the other hand, are static and are managed as a centralized, homogenous, symmetrical framework of visibility and interception. Even though access to the data collected from such an environment provides some accessional improvement to researching new algorithms and detection methods, these incremental offline advancements are vetted in a sterile, non-real-time environment without the benefit of sequent responses or adaptive determinations accoutered by a production environment. The primary goal of the Defense Research Engineering Network Cyber Security Test Bed is to leverage emerging network protocols and recent distributed computational techniques to create a cloud of sensors built on tractable computer server platforms that enables cutting-edge security to coexist with current security infrastructure directly inside the production network. The transition time of the latest cyber research from theory to practice will be significantly reduced while intrinsically revolutionizing the approach to engineering network security architectures. By creating a true proving ground by which the science of new algorithms and detection methods can interact directly with raw (as opposed to filtered, sensed, or captured) traffic in real or near-real time in a safe and controlled way, the proposed test bed will provide meaningful advances that can appreciably address the ever-changing landscape of cyber attacks.

Key words: Adaptive; cloud; Defense Research and Engineering Network (DREN); detection; live; network; security; sensor; test bed.

It is an easy sell, at least on the surface, to discuss the introduction of new detection and protection techniques into the traditional network-centric security posture. Just beneath the surface, however, it is clear that traditional architecture has been engineered with specific requirements in mind and purposefully uses a relatively simplistic model: make it persistent (or make it static), consistent (centrally manage and control it), homogenous (use the same tools everywhere), and symmetrical (see both sides of all connections) wherever possible. The solution is the result of a steady evolution from deploying detection and protections in key locations on a network boundary in a Draconian fashion based on event history to modernizing by adding layers of automation, such as firewall or intrusion protection system (IPS) signature subscription services and domain policy for maintaining system vulnerability patching for up-to-the-day readiness. Practically, a circumscribed "defensible" perimeter represents both a sphere of visibility and control and a clear demarcation of responsibility, while reducing the complexity of management of enterprise-wide security solutions and systems within the boundary. Therefore, introducing complexities into that equation can easily translate to more manpower and effort as

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2010		**00-00-2010 to 00-00-2010**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Joint Sensor: Security Test and Evaluation Embedded in a Production Network Sensor Cloud	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Defense Research and Engineering Network,High Performance Computing Modernization Program,Lorton,VA,22079**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**13**	

well as the opportunity for errors that can easily hide real problems.

In contrast to traditional defense, today's threat is dynamic, decentralized or distributed, heterogeneous (using various means and methods and attack vectors), and asymmetrical. It connotes a style of using a diverse platform to launch a barrage of threats and even turn the defense mechanisms against themselves to deny service, compromise systems, malign applications, exfiltrate data, and then (as if that were not enough) use those systems to launch subsequent assailment—all without being detected firsthand and often not until long after the event. It then comes as no surprise that efforts are now placed in a variety of avenues to aid in identifying malicious content, anomalous traffic patterns, and even behaviors of the programmers developing malware. The security architecture that takes advantage of all of these developments by delivering more comprehensive and compatible features that improve detection, investigation, and mitigation is far more likely to yield the significant gains required to remain afloat in the face of cyber storms.

It is not, however, as simple as replacing one paradigm with another or one tool with a newer one. It is primarily a philosophical transition from the historical function of security to seek out and categorically block incoming antagonists to a more surgical and focused reaction to maintain as much operational normalcy as possible while defeating specific intrusions. In making that transition, tools cannot necessarily just be upgraded in place or abandoned for newer ones. The nature and function of the newer tools rely heavily on hardware and processor capabilities and fit into more fluid data communications structures not entirely compatible with the existing installed base of components, protocols, or data formats. Likewise, research and development of these new tools and techniques using only data that is captured using the traditional methods may also lack the perspective to uncover new tactics and unleash new response mechanisms. Instead, there must be a way to leverage existing capabilities to create measures of usable data, as well as grant access for the next generation of detection and defense algorithms to be proven within the current architecture with real and real-time data to smooth the transition and transform the defense strategy.

It is precisely this gap that a neoteric sensor platform can help fill by injecting research methodologies and tools into the existing architecture. A replacement for traditional sensing appliances, this solution combines current sensing techniques with an isolated modular space within which to test new tools and strategies on a multipurpose hardware platform directly within the production environment. The goals of such a device are to continue to provide existing capabilities, enhance those capabilities with small doses of new techniques for detection and protection, and significantly reduce the development cycle from research to production for quality tools. These nascent methodologies must be implemented without breaking performance or compromising operations and be directly subject to the same inimical traffic to both better sense the anomalies initially and provide clear value by uncovering threats and activities not previously detected.

Divergence of attack and defense style

Many security professionals feel the approaches for cyber defense of the past need to be amended or augmented to find new attacks so that we may continue to "meet mission in the face of cyber warfare." Alternatively, some people go further to identify cyber warfare as a fifth combat arena (*The Economist* 2010) behind land, sea, air, and space. The latter camp builds an argument by identifying three unsound assumptions with the former camp:

- The boundary is structurally defensible (which does not account for mobility),
- The threat is more readily tractable on the existing dimensions being defended (in the face of multifarious attack),
- Automation is equivalent to readiness (which both relies on an asynchronous mechanical client update system and intrinsically trusts the content and structure of resultant code).

A read of this year's Verizon 2010 Data Breach Investigations Report (Baker et al. 2010) may in a sense reiterate the assumptions and propagate the impression that the most significant issues in cyber security are resolved by better deploying the existing technologies. Their statistics, generated in collaboration with the U.S. Secret Service as a study of existing cases of breaches, indicates that 61% of the cases were discovered by a third party, 85% were not considered difficult to accomplish, and a whopping 96% of studied breaches could have been avoided with the use of some form of low- or median-level mitigation. The conclusions were based on one glaring fact: in 86% of the cases, victims had evidence of the breach in their log files. Taken at face value, this type of study shows the need for bolstering and continuing to emend existing installed solutions that mitigate known vulnerabilities, but it does not necessarily mollify the need for a broader perspective on warfare or keeping even worse from happening. Perhaps more germane to the argument is that the method of the study might indicate a marginal disconnect or divergence between

defense and attack style simply because the data being used to develop conclusions, or in turn, new tools, was uncovered using conventional data and visibility.

Though the Verizon report shows hackers still have open to them several "paths of least resistance," simply closing those paths does not secure against the broader, more organized, mature, and insidious threat that cyber warfare proponents assert. And while the report further catalogs the expanse of the threat, the conclusions are focused on closing the known holes and implementing processes to look for more holes. While this retrospective admonition has merit, efforts based on this distractive construct of integral improvement not only diverge from the attack style but also propagate the limitation to invest in and support development of modernized capabilities. In gist, what is being asked is to fundamentally change how defense is enacted. The real imperative of cyber warfare is to defend against something of which we have no knowledge, arriving as a previously undiscovered zero-day attack, and entering on one or more vectors about which we do not know and over which conventional defense has little control or visibility. Even so, outfitted with the latest weapons on each of the vectors, the strategy focused on finding something new to block may in practice allow the attacker to use defense systems and practices against themselves, whereby an effort to block the traffic in a gross motion may result in a self-inflicted denial of service. Summarily, war in the cyber realm contends for the need to stop just defending, bring the various vectors into a unified interdependent defense model, and have concentrated reactions specific to the attack. Further maturity is then needed in a strategic shift from force protection to surgical response with focus on precision mitigation, low false positives, inoculation or learning where possible to defend against repeat or similar attacks, and effective and immediate recovery of systems that have been attacked.

The mobility and dimensionality notwithstanding, with the perceived necessary central approach to manpower, management and reporting, ease and consistency of tools (i.e., homogeneity), and automation (e.g., updates and filter list managers), the detection and pragmatic block of all possible mechanisms and signatures to avoid any potential known types of attack simply is not scalable. Antivirus powerhouse Symantec announced in its publicly available quarterly report that it created 457,641 new malicious code signatures in the second quarter of 2010, down from 958,585 in the previous 3 months (Symantec 2010). McAfee indicated earlier this year that growth in new malware recorded remains around 40,000 pieces per day (Muttik 2010). (How many signatures can an IPS run before a significant drop in performance occurs?) The signature convention of blocking all possible inroads based on historical attacks is unsustainable. Moreover, these staple products use an asynchronous method of update, whereby the new malware must be identified, submitted to the vendor for processing, and then downloaded, with a periodic client update to be installed on a computer to detect and then possibly mitigate a future infection by the known malware. Considering the incredible capabilities of worms and other attacks like Conficker and Aurora, perhaps a deficiency in the current model that is even more sobering than scale is time.

By the time the press had come out about Google being under attack by Aurora in January 2010, at least 34 other organizations had indicated they had come under the same attack. Ongoing worm research has also indicated that, through similar techniques, as many as 1 million hosts can be compromised by a worm in as little as 0.5 to 1.0 seconds (Stanisford et al. 2004). So even the most robust traditional security just does not scale to protect from initial attack and does not learn fast enough to keep the compromise from spreading. Details now known about the Aurora attack indicate that the attack was so diversified, used encryption and obfuscation techniques of a complexity not seen before, and came from such a spectrum of sources as to avoid traditional detection. In just these few examples, it becomes clear that time is not on the side of traditional processes and even the best of traditional security is not geared to detect, let alone respond to, the changing threat. To wit, after further study, botnet expert David Dagon of Georgia Tech provided a telling rejoinder that beyond the network-based security not preventing the spread, "the network *is* the infection" (Dagon 2005). Therefore, the defense style must, by virtue of functionality, now be transformed from an irresolute, static, and isochronal response to a dynamic, flexible, and predictive one to detect sooner and more effectively thwart the ever-changing attack.

More to the point, the evidence on all fronts drives home the requirement to bolster the advancement or replacement of traditional tools aimed at dynamically updating host defense and anomaly recognition, prime the ontogenesis of detection and containment techniques, and incite the discovery of new tools and algorithms designed to see new kinds of attacks before infiltration. Those all begin with access to data. Typically, new tools are developed using simulated data, network replay, or analysis of collected sensed data combined with data stores of log files and similar system-specific information. Any systematic approach to innovation by requirement features access to that

data, at least as a tenuous first step in progressive ingenuity. However, storage directly affects capability (i.e., you can only store so much for so long) and retrospective analysis directly affects network bandwidth (i.e., all this sensed data must be backhauled to a central location for processing). Therefore, even in this first phase, generally any data collected must be limited by some suspicion threshold that triggers capture before caching locally and then compressing for transmission to the depository. Care must be placed on the right rule sets to balance the amount of data being captured and the bandwidth and storage required to retain it. (At what decay rate does the data collected from such attacks become unusable for improvement or development of new strategies?)

The other shoe

"Botnets" are not just quite prevalent today but all the rage, creating a marketplace that is both highly sophisticated and inexpensive. It has been reported widely in recent months that hackers are having what some call a "fire sale," whereby an interested party can buy a "botted" computer for a slice of time for as little as $0.02. That creates an inexpensive attack infrastructure that is not only voluminous and widespread but also highly adaptive and dynamic. Imagine being an upstart hacker needing to test a new algorithm, distributed denial of service, or a malware solution for data exfiltration from a company or even a federal agency. Being able to rent a large infrastructure of botted machines from around the world for the price of a few hamburgers would surely facilitate a large-enough attack source or malware hosting facility with sufficient obfuscation as to provide immediate results on the validity of the code while avoiding detection of the actual traffic going into and out of the Internet access gateways, let alone tracing of the sources or criminals moving as quickly as the shadow of the cloud under which they hide (Delbert et al. 2010). Further, the high availability and variety of different systems in diverse locations make it possible for hackers to rent the appropriate facility to reduce the hacker test-to-production life cycle for their malicious wares. The proverbial shoe is clearly on the other foot.

In stark contrast, researchers in retrospective-based defense system development rely heavily on large, expensive, summarily classified or otherwise unavailable (and as a result, often quite stale) data sets with rigid posture and limited scope to test their ideas. Trying out new tools is not only complicated and time consuming but also costly and often delayed enough that researchers cannot know whether the ideas will bear results until just before or even after the finished product is sent to market. Even more likely, the research takes so long and costs so much money that the resultant technology is already stale or unusable and is insufficiently up-to-date with the malicious world against which it was being designed to protect. Behavioral heuristics (detection of traffic anomalies), code obfuscation and encryption detection, infection detection and quarantine, and cyber genetics certainly all have value and are being funded and pursued more than ever, but all face this same dilemma. Without early testing of ideas on quality offline data, intermediate validation of budding algorithms using increasingly real-time traffic, or full-scale evaluation of the resultant solution in real time in a real environment, the cycle from idea to production expands to a nearly untenable and mostly unsustainable ambit. It then seems relatively imperative to find new ways to get data to the science or, even better, get the latest science more fugacious access to the data to bring that science to market faster. Further, engaging the mature tool sets in this final stage of access to contextual traffic for evanescent validity, the construct is finally broached for ongoing support of the research, stimulating both evolutionary and revolutionary adaptation to the threats while improving the effectiveness of the installed security base.

Newer defense systems generally begin to abandon signatures that require mechanical updates to a large database running on the appliance supplanting them with software or even Application-Specific Integrated Circuit (ASIC)–based algorithms that look for particular behavioral patterns in the traffic or anomalies in the data. Detecting certain ways a malicious code will try to behave to avoid detection, phone home, receive remote instruction, or seek to further infiltrate are more and more known by analysts and researchers to the point of guessing that if that pattern of traffic occurs, it may well be malicious in nature. Some researchers have gone further to indicate that programmers operate with similar behavioral patterns and are more easily detected. The desire to remain anonymous, to fragment traffic to avoid header analysis and the sheer numbers of resources around the world to enlist as an attack source are all clear indicators of a need for suspicion, and inclusion of these characteristics reduces false positives significantly. In some cases, these kinds of indicators are being built into the tool sets scanning data in retrospect and into IPS and content scrubbers watching traffic patterns, but in other cases, it is a unique research algorithm searching public information for human behaviors and configuration patterns. For example, how a Domain Name Service (DNS) server is configured in support of a domain might indicate the administrator's desire to

remain low key. A broad study of publicly available DNS information may uncover a trust model or likeliness of becoming malicious at a later date. Cordoning off traffic to and from those addresses, domains, or network neighborhoods for more intense scrutiny would be more palatable because it reduces the amount of traffic that requires such scrubbing and the hardware and bandwidth required to watch with intensity.

This particular type of discovery is not exactly easy to do but does not necessarily require access to live traffic. Yet information being learned from such research becomes a more critical part of the overall protective mechanism and must be integrated into the comprehensive defense posture. Were the shoe on the other foot, or more precisely, were the shoe in the production environment designed to assimilate or accommodate the other shoe being developed across multiple research communities, the ability to test the capability on the operational foot would provide profound help in bringing both capabilities to bear on protecting the network. Unfortunately, the current architecture is often device based, or even ASIC based, and introduction of new algorithms and tools is not simple. And incorporation of the solutions others have found is even more incoherent (reminiscent, to maintain the allegory, to the intelligence fiasco of *Tall Blond Man with One Black Shoe*). Instead, research from various fields and intelligence gained from those fields need to be married in a new approach to dynamic security posture.

In complement, several security vendors, as well as federal institutions and federally funded university institutions, have made significant strides in Internet Protocol (IP) trust models. By using both collected data from a worldwide installed base of sensors and firewalls, as well as security alerts and massive data stores of attack data, researchers have been able to create incredible databases used to categorize bad agents, agents who may work with bad agents (guilt by association), and hosting or network providers and supporting bad agents (autonomous systems that provide a safe haven for agents that generally do bad things). Cognate to these approaches, innovations such as Milcord's Botnet Threat Intelligence has been able to identify malicious content providers by rapid changes in DNS information or the hosting of large numbers of domains across a very small number of IP addresses, a behavior known as fast flux (Caglayan et al. 2010). Colorado State University's Border Gateway Protocol (BGP) Monitoring System (Yan, Massey, McCracken, and Wang 2009) has vastly improved the capability of identifying suspect organizations based on route changes found in BGP route tables from around the world. And there have been several ventures into genetic and immunization models for dynamic defense, worm quarantine, and other means to identify and prevent the spread of malicious worms and Trojan-based attacks. These efforts, and others as they arise, need to be verified in the real environment, as well as incorporated into the overall attack strategy of the enterprise.

The Defense Research and Engineering Network (DREN), being the wide area network service provider for the Department of Defense's Research, Development, Test & Evaluation community, is in a unique position to collaborate with these types of research in pursuit of the development of a robust and dynamic security posture servicing a mature and complicated environment. This solution is principally delineated into five elements:

- Identify (mainly through intelligence) historically or potentially malicious players (some through behavioral analysis), networks, or "neighborhoods" before they attack, and handle traffic to or from those entities appropriately;
- Detect or sense suspect traffic patterns as early as possible to identify and contain events as they are under way;
- Combine near-real-time alerting and response (or countermeasures) with retrospective analysis to get to quick reactions as events occur (or are suspected to be occurring), as well as full details shortly thereafter;
- Use dynamic networking capabilities to create a tiered approach to protection and prevention, creating a mechanism for best-in-class protection being provided by different devices, potentially even in different locations;
- Facilitate the research in all these areas by providing access to data with appropriate controls and platforms so that new solutions can be proven and brought to bear sooner.

Through collaboration on several diverse projects with other federal and federally funded agencies and programs to pursue each of these elements, DREN can readily vouch that significant research has already been done in many areas such as improved analysis, heuristic-based detection, and even behavioral analysis of the attacker, all to provide increased intelligence of who the bad guys are, where they tend to live, and what they look and act like. Those data points and projects have increased and continue to help increase the vital ability to defend the ever-changing environment in pursuit of an ever-evolving mission. Integration of these research elements, by redeveloping the architecture to make it compatible with influx of intelligence

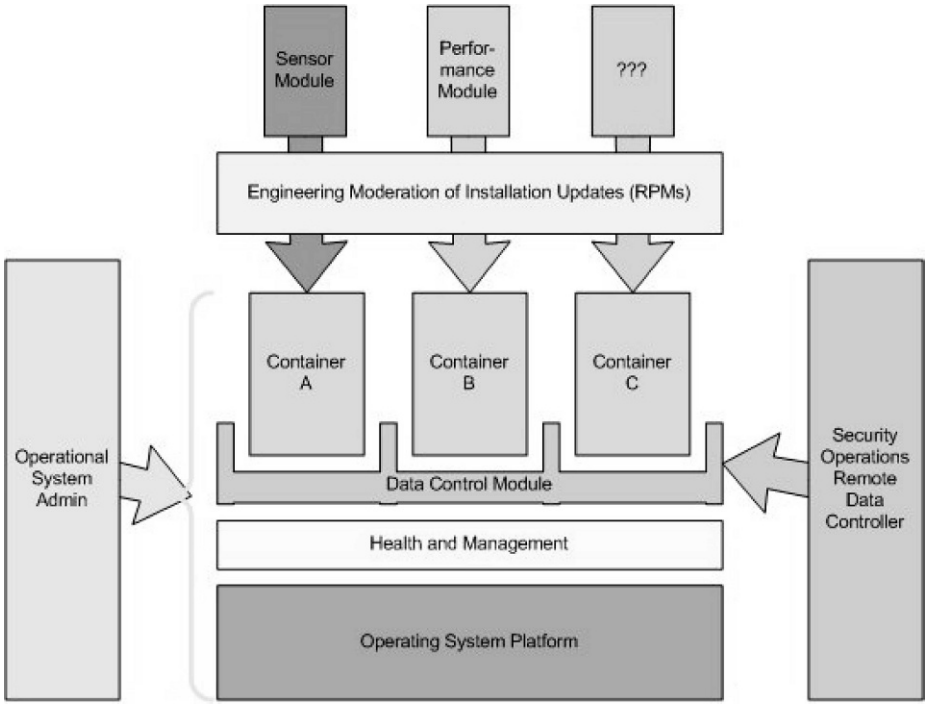

Figure 1. Joint Sensor functional architecture.

and rethinking the data structures to support analysis from both internal and external sources, provides the pivotal pieces to migrate the traditional architecture in pieces to a more robust and nimble posture. Being a wide area provider, however, puts DREN in a unique position to focus our efforts on the final of these elements, namely, getting data to offline scientific research and getting the results of that science into the environment for proof and innovation. For this reason, the Joint Sensor (JS) project was initiated to concentrate our proficiencies for the betterment of cyber research in pursuit of that grand defense solution.

The DREN JS

The crux of the DREN security architecture that spans all five critical elements of strategic defense is the invention of a new multipurpose platform, the JS (see *Figure 1*). This platform services both real-time and retrospective analysis tools, will be fed in places by dynamic network redirection of suspicious traffic identified through intelligence feeds and other mechanisms, and will make access to data more readily available for proving research in a real environment. While the main idea being conveyed is to allow testing assets to sit next to or on top of a device already providing production sensing, the JS platform is more broadly valuable in that the platform is capable of supporting some of the more rudimentary requirements posed by a security operation with a dynamically changing deployment strategy—namely, full or partial reimaging, traffic-specific sensing, and functional participation in a cloud-based distributed environment.

The key to the JS project's success over the various historically incompatible uses (operational, data mining, and research algorithms) is its component design as an appropriately sized, multipurpose computing platform with data connectivity allowing for the collection of sensors to act as a fully capable, distributed, node-parallel computing platform. To gain the advantages of this, various methods of data separation and protection must be applied internally to a system and then externally to how they are connected to the network. The system's specifications were also selected as being able to support the more recent advances in virtualization and process isolation, making it possible for the various elements to coexist without treading on one another. Then, in addition to having enough capacity to operate correctly when at full network flow load, the design of the system allows for other uses when the network load is normal and far less than full capacity.

Computational techniques and hardware for a given price point have advanced far enough to apply better than simple signature techniques to improve cyber security. Given the multicore, large memory system that is required to deal with a high-bandwidth or denial-of-service attack, a significant amount of processing power would be available at all other times

for nonoperational uses. Classic design of sensing requires many sensors located at the network data. This means that the JS systems across DREN mostly will be available as a large, distributed-processing computing capability. Advances in scheduling, memory use, data mining, and parallel techniques would allow this capacity to perform research and test functions without interfering with operational function as a classic security sensor. A key advantage to this approach is that the same network flows (Rajahalme, Conte, Carpenter, and Deering 2004) that are being examined in the normal ways can be examined, mined, and vectorized by both emerging applications and research sampling methods.

In addition, with the bandwidth available at off-use periods on the high-performance DREN network, both existing and new methods in the area of data cross-correlation can be researched and developed as a stage toward integration with a response mechanism. When combined with tools such as BGP Flow Specification (FLOWSPEC) and other network data copying and redirection techniques, network flows of interest can be sampled or piped through other resources such as those available from the High Performance Computing Modernization Program for even more advanced and intense algorithms. These algorithms, working on live data that is representative of both normal and intrusion-type flows, can lead to new techniques of detection, elimination, and even potential cleansing to deal with the ever-changing threats. Even in the area of existing operational sensing, the JS project can add a new capacity by providing communications and computational frameworks for doing simple distribution and redistribution of signature and threat analysis processing across all nodes, shifting work from heavily used systems to barely used ones. Techniques can be applied to cross-correlate data and findings across all nodes such that, for example, network flows seen in more than one location are only processed or analyzed once in the path. Moreover, signature hits can be multicast to the other nodes to increase monitoring of related flows or to change scheduling in recognition of a high load event spreading across the network.

Data separation in the joint sensors will have many facets. Initially, the standard mechanisms are process core binding and data isolation, combined with hardware-supported memory protection. The operating system was selected to be able to take advantage of securing mechanisms available now or being added incrementally, such as security-enhanced Linux, containerization, and full virtualization. Since a large reason for both the operational and the research use of a JS system is to capture the full network traffic at the location, a mechanism will be developed to act as data controller for that network capture. A single capture will be passed to one or more existing modules or methods of analysis. A more controlled and possibly restricted copy of the data would be made available to research algorithms on the system. A sanitized and restricted data set could be made available on the system (most likely to a separate virtual machine on the system) for use by external and affiliated researchers. In addition to all of these methods local to the system, the data controller could send a full or subsampled set of the data using an encrypted path to other resources for further analysis or research algorithm processing. This last method could also be used to make diagnostic captures based on a filter definition fed from a remote, fully authenticated control station. Similarly, controls could be passed into the system, such as to satisfy data requests from law enforcement, which could include instructions for the data controller to apply special encryption to the data and pass multiple copies to distinct and appropriately controlled archival systems.

All processes will be fully vetted before deployment on the JS. In addition, extra steps can be deployed to ensure proper function even while running a research module. Some such modules can be subjected to additional memory and processing restrictions (core affinity, central processing unit utilization, and memory allocations), as well as techniques such as memory leak monitoring and process destruction, to ensure no deleterious effects on the mainline processing of the system. In addition to these mechanisms within the operating system, new techniques provided by libraries and processor features will add virtualization capabilities to provide further isolation. These methods include containerization, which allows a process to run on the main operating system but with no access except as defined to the operating system and leaves the process unaware and incapable of interaction with other processes on the system. A further step would be to do full virtualization, where a module would exist with its own operating system and copy of the data without interaction with the host operating system, any process on that host system, or any data or process of another guest virtual machine. This technique also allows for any Intel-based operating system or appliance-like package that is completely different or incompatible with the RedHat Enterprise host operating system to run locally and have the captured data set available (using internal host or guest network interfaces).

These systems are homed to the DREN network. By its mission, DREN is a high-performance, high-capacity network to transport Department of Defense research and development, science and technology, test

and evaluation, and modeling and simulation data. As the transport and security of the data results of the sensor function need to be assured, this is another place where separation and control techniques need to be used. The DREN architecture provides several mechanisms that are useful to this need. First, multiple layer-3 IP virtual private networks can be used with varying amounts of separations to ensure connectivity and monitoring of the JS system; and delivery of its data can be handled in a separate and preferentially queued way. The path of data to CERT operations and internal research systems would use this method. The redirection and cloning of data using BGP FLOWSPEC techniques would also use similar layer 3 methods available. At the next level, DREN can provide isolated layer 2 paths. Using virtual private local area network service in a configuration developed for another project on DREN, the console implementation will use a layer 2–separated path from the operations control points to the sensors while not allowing traffic between the sensors. This console implementation provides connectivity to the onboard integrated Dell Remote Access Controller interface, which implements a full Intelligent Platform Management Interface 2.0 capability and then some. What this means is complete control of the system from a remote location with console functionality—both serial and graphic, as if locally connected—but only from predetermined locations. In addition, this has the capacity to mount a remote image that appears as if a digital video disc was inserted into the system.

Using this combination of tools, a complete system boot from the remote image; preparation, install, and customization of the operating system; and inclusion of all add-on modules can occur across DREN in a private, controlled manner. This can occur remotely in about 35 minutes, in contrast with 25 minutes when using local media in the local installation lab setup. This capacity not only provides full control to ensure that the sensor remains fully functional at its operations mission but also allows it to be adaptable and even completely remoldable without significant shipping costs, travel, distributed manpower, or downtimes that are longer than necessary. Since this is a limited virtual private local area network service deployment of layer 2 connectivity, its separation from any other DREN function is high, and the locations with access to this remote console capability can be tightly controlled to restricted, on-network sites. Using the system's other interfaces, additional paths will be set up to manage the system, collect data for the operational CERT functions, communicate with the other JS systems, and have the capacity to set up temporarily unique captures and data paths over DREN. Having these interfaces and paths allows the use of the features inherent to DREN to provide high-capacity, secure communications where data protection and integrity are ensured.

Expectation engineering

The final piece of the sensor bed puzzle is engineering the willingness to support such an intricate solution. The success of even getting such an emergent test bed deployed within an operational environment boils down to three key elements, at least in terms of bringing to bear the right framework to create and sustain the environment, as well as to provide sufficient verdure to attract willing parties and sustain harmonious living within it. These three elements are as follows:

- Providing access for the researchers to real operational data sets (traffic, data store, central storage, or other appropriate capability, whether on the device or in a controlled shared space), as well as to the test sensor packages in the cloud for managing and making changes to the product;
- Indirect but immediate sharing of algorithms to security operations that provide visibility into attack vectors not otherwise seen using traditional sensing and showing intrinsic value for the arrangement;
- Guaranteeing some level of control but, more importantly, significant levels of visibility for network and security operations into the function of test capabilities and the process whereby the first two key elements are managed and delivered.

In an environment where security is prime, there has historically been a separation and isolation between operations and research, usually upheld in reality by dividing activities on network segments (e.g., demilitarized zones, border zones, and lab networks), as well as temporal separation between live traffic sets and data being offered for research. Simply put, there has been limited access to the live network by anything other than stable and secured applications and devices. To facilitate this test bed and provide benefit to both operations and research that are nearer real time and lasting in effect, some of those conventions need to be broken down, and the research must be meticulously inserted to maintain the original character of security. At the outset, this translates to not only best practices but also sophisticated operational security measures on the joint sensor, as well as in the processes of securing the applications before use. To make network security operations a willing participant, these additional functional requirements for the sensor test bed mean leveraging stable and tried operating systems, middle-

ware, and application configurations in the field. Perhaps far more important to success and willingness, there needs to be visibility access granted to security operations—not just into the additional software but also into the processes and procedures of how those elements are managed and maintained. Controls must be implemented on how information is shared with the software and, in turn, from that software to its management systems, source coders, and stakeholders. But more rudimentarily, this leads to an emphasis on visibility into the process of fielding a package into the sensor test bed so that operational security can inject reviewing hinge points and affect policies at various review stages. An engineering resource internal to the organization that will take the research participant, along with the operations participant, through the process from concept to field trial gives all parties sufficient voice to ensure the solution is engineered within expectation and guidelines.

This progression is also a phased approach, whereby the research participant begins with access to sterile data with which to run algorithms to do a rudimentary proof of concept offline, followed by an experimental initial offering in a lab environment using real but not real-time data, quickly proceeding to a similar scenario where live or nearly real-time streams of operational data are tested for verification of algorithms, as well as constructive processes such as management and alerting. These earlier phases give the researchers the opportunity to test their theories before expenditure of operational man-hours and resources for field deployment and to create a more trusted expectation once field testing is approved to begin. Through this phased process, another of the key elements is awarded participants: algorithms and actual results using live data can supply researchers with validation of the algorithms and demonstrate to operations with evidence that these algorithms are of value. Once in the field, a more intimate relationship between researchers and operations (or at least the output of the test sensor and the input of the operational security mechanism), brokered by the internal engineering capacity, will give the security participants more immediate value by finding issues their tools would not otherwise have found. Conclusively, these algorithms, running in parallel to existing capabilities, provide a number-for-number cross-correlation of results, false positives, and detection rates as all are subject to the same traffic.

Ultimately, the grandiose concept of a cloud of test sensors built on the back of the production sensors requires as much operational nuance as it does technical innovation to ensure environmental policies are enforced, participants have valid expectations, and real results have both immediate and long-term impact. In the DREN JS project, the supporters of the former methodologies have been enlivened by the opportunity to provide input to the building of the sensor and the process whereby the sensor will be managed, and part of that enlivenment was directly created by an enlightenment of seeing new sensor technology find real issues in data that the incumbent technology had not seen. Billing the new technology not as a replacement of the existing methodologies but, instead, as an enhancement to them would not have created willingness without the proof of real data and visibility into how the system would fit into the architecture.

Dynamic network support

In general, the effectiveness of intrusion detection systems, intrusion prevention systems, and even firewalls is like real estate—location, location, location—and in general, prime location is at the enterprise edge facing the wide area network. In the case of DREN, a wide area network service provider, the edge is an asymmetrical collection of geographically diverse network access points connecting the network to a variety of upstream and downstream entities, such as tier 1 and regional Internet service providers, direct and private peers, and of course, customers. This renders the key location to see the most traffic a less-than-optimal location to see both sides of any given conversation. As best current practice for a wide area network is still "hot potato routing" (getting packets off your network as fast as possible via the closest connection along the path to the destination), the capability to synchronize bidirectional conversations is impractical and nearly impossible to manage. However, significant capabilities in network equipment can now facilitate "symmetrizing" predetermined connections or "flows"[1] such that the traffic to or from a particular prefix or protocol of interest can be dynamically redirected to a remote-triggered black hole, sink, or scrub (see *Figure 2*).

A remote trigger is a means to dynamically tell a network device (usually a router) to redirect traffic with certain parameters (source IP address, destination IP address, IP protocol, and transmission control protocol/user datagram protocol source and destination port) as it hits a filter or access list, sending this traffic to a black hole (a means to drop the traffic), sink (a collector designed to capture the traffic for analysis rather than just drop it), or a scrub network. A scrub is a collection of tools that can be in line at a separate location somewhere in the network cloud used for monitoring or performing stateful inspection, intrusion protection, or content filtering, and allowing valid traffic to proceed unchanged to its original destination.

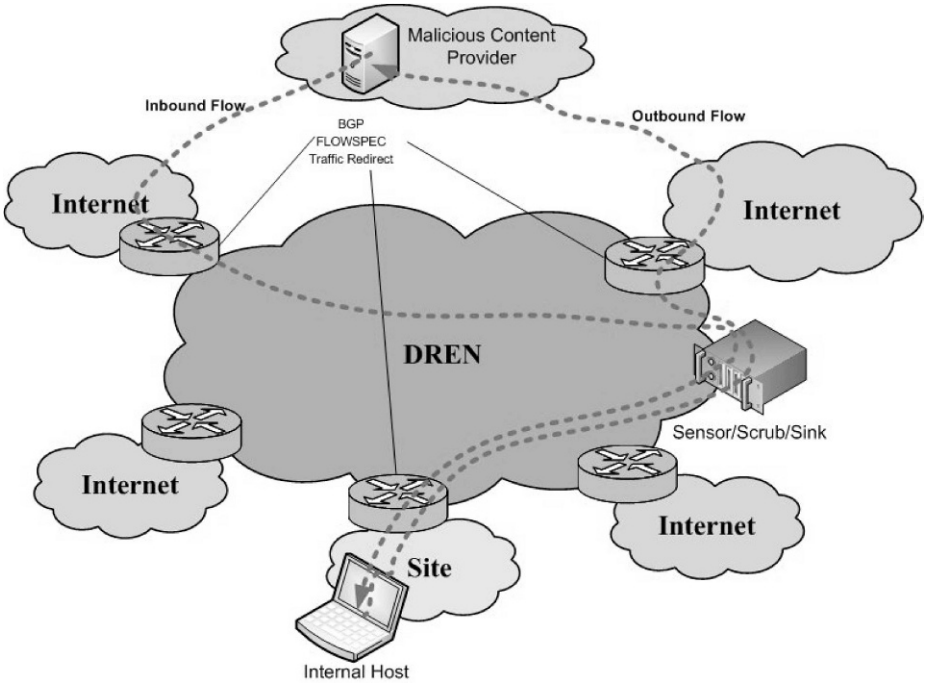

Figure 2. Dynamic redirect using Border Gateway Protocol Flow Specification.

This scrub network redirection is facilitated by configurations and protocols designed to temporarily modify the path of the traffic within the network without giving any indication to external entities that the redirect is occurring. Even better, because the network can be affected bidirectionally, traffic through the scrub is now symmetrical, and both sides of the suspect conversation can be monitored through a single inspection point.

Because this redirection is now available, it is no longer required that you have all the right tools at every possible location, and tools at specific locations can be specialized to focus on particular protocols or traffic types. For example, consider a subscription service to malicious URLs or an intelligence feed of alerts about suspected "botnet" addresses. It is possible to inject this intelligence into the network so that the boundary devices (facing the ISP and the customers) upon receiving any packets associated with these suspect addresses and/or protocols can be redirected to the scrub (or sink). There the tools can now see both sides of a connection in order to accurately determine malicious content. For the purposes of a sensor test bed, you no longer are required to have sensors at every site to be of significant value. An algorithm that focuses on malicious web traffic, email scanning, DNS attacks, or any application-specific determinism can now sit in a single or a few locations and achieve complete visibility into all interesting traffic for focused testing.

One particular element, a new network layer reachability information protocol of BGP known as FLOWSPEC, allows a system (such as an sFlow collector or analysis tool) to publish "rule sets" in the fashion of particular flow or traffic parameters in a BGP session with a specific action (such as discard or redirect). This gives the security operations the capability of dynamically updating filters on routers across the wide area simultaneously. Used in conjunction with virtual route forwarding tables and protocols like multiprotocol label switching, FLOWSPEC enables dynamic blocking—or better yet, redirection—based on information gained from outside sources, deep packet inspection, or retrospective analysis. In conjunction with a sensor test bed, particular traffic patterns of interest can be "symmetrized" and sent to a set of tools for better isolation and tighter focus of research algorithms. In the context of the phased approach mentioned earlier, dynamic redirect could surgically separate known-suspicious packets and send copies or temporarily divert that traffic through a controlled, laboratory-type environment without having the test software residing in the production environment or touching valid, sensitive data.

As an aside, this capability can change with the perspective of the researcher. As indicated earlier, it is possible to redirect traffic based on an intelligence feed, the outcome of internal analysis, or other such means of identifying peculiar or suspicious traffic. Likewise,

just this part of the overall mechanism is a means to research emerging algorithms or tools in that the choice of what traffic is diverted to a particular sensor or scrub center may be determined by the work of a new tool or even an outside research project. Today, feeds from Milcord, security vendors, lookups from the BGP Monitoring System, and others are all possible candidates for a redirect to an algorithm-specific sensor or protection system, making it readily possible to divide and conquer and thus reducing overall performance and bandwidth requirements on any individual system and constructing a defensible boundary one protocol or application at a time. Other methods for identifying things for redirection can also be tested safely, including research such as cyber genetics, man-in-the-middle botnet investigation, and immunity algorithms for identifying bad patterns and other negative characteristics. This thinking goes quite well with the distributed parallel computing capabilities of a collection of joint sensors across DREN.

In the end, dynamic network support is critical to facilitating the next generation of traffic protection, sensing, and enterprise-wide dynamic security architecture with a focused attack response. In the meantime, for the purposes of a test bed, it becomes quite powerful in facilitating the first phases of proof of concept for new algorithms and tools before they are put in the production realm. This extra step in the process provides sufficient "warm up" time for the security operations teams to assess or remove any question before putting any risk on the network. Similarly, the ability to send only the traffic that needs to be seen by the particular algorithm, selected specifically for its suspicious or known malicious nature, means the tool does not have access to sensitive data but does still get sufficient real and real-time traffic to perform the research. Any anomalies detected or protections proven through this dynamic redirect give the research considerable value and provide security operations tremendous insight into the function of these new tools—without putting them in production.

Next steps

The first opportunity to improve the development cycle of interesting new tools from concept to production is to provide data sets to researchers for early analysis. Once the algorithms that are implemented in the tools are proven and improved through access to production traffic, the logical next step is to develop a means to incorporate what the algorithm detects into the overall, aggregated analysis and response system of security operations. Systems and algorithms developed through research around the world result in new intelligence feeds and alerts that can feed the central aggregate analysis in the production environment, as well as those rule sets indicated in flow data and BGP FLOWSPEC deployments. Likewise, these tools being developed and possibly deployed as products in other networks and research environments should then result in new alert feeds made available to this network as production tools. The aggregation of these data streams is critical in the next generation of security architectures.

A project under way at the Naval Research Laboratory in Washington, D.C., is taking this concept and creating a sort of cross-correlation system. Tools such as host-based security systems, firewall and IPS databases, malicious uniform resource locators, and other subscriptions are all being synchronized to create a multidimensional set of target parameters. DREN has a similar function being developed, whereby a scripting system is used to indicate whether particular IPs are showing up in multiple intelligence feeds. Certainly, any IP or prefix or autonomous system number that appears in multiple lists should be regarded as a more serious threat and can be more closely scrutinized. In addition, taking data from multiple solution sets from various vectors can help create a richer attack vector analysis and present analysts and dynamic watch systems with a sort of trust model of dangerous protocols or "bad neighborhoods." Just as long lists of individual known bad IPs is hard to digest and incorporate in a watch list or analysis tool, too many tiny fragments are that much harder to distribute through protocols on the network. Therefore, being able to combine knowledge from multiple sources and different types of information into a general attack and protection pattern simplifies the architecture and provides a more robust response system.

As previously indicated, dynamic protection also engenders the focused response as a measurable outcome to be researched during this project. The researchers that are creating the algorithms should be working in concert with the security professionals to develop response mechanisms and methodologies as part of developing the detection and prevention strategies. Mitigation recommendations and progressions for various traffic types or attack intensities, reduction of false positives, inoculation against repeated attacks, and means whereby infiltration can be recovered from must be incorporated into the defense strategy as each technique for discovery is pursued. Expecting the vendors and researchers to provide both new alerts and sustainment is critical in the thought processes required in making a marketable product, as well as an integrated tool for use in the environment.

One next step in the sensor test bed project is to develop the mechanisms and processes whereby potential suitable research initiatives can begin taking advantage of this new solution. These must be finalized and put into a quality management system. The goal of this programmatic development is to better understand the nuances of how to select the most valuable and most mature tools first and get immediate benefit from the program. In parallel, as a measure of effectiveness of the program and the tools it produces, we must also focus on improving the political relationships and creating solutions that are more immediately usable by the broader federal community, even while still in the infant stage. The tools being produced through the test bed should not be limited to use within DREN and should also promote the development of (or compatibility with) community-centric capabilities such as shadow-mirror databases, standardized data structures and formats, reporting templates, and alert communications systems. A tool being developed through the JS test bed may bring to bear components where national or vendor-specific collections of attack signatures and threats—like McAfee's IP Trusted Source or Symantec's AV database—would benefit the community greatly and much earlier than traditional procurement processes afford. A shadow-mirror is a duplication of a vendor database that receives updates from the vendor system (being populated by submissions and alert feeds from all over the world) but then becomes the internal collection point of new additions discovered in the wild within the community (rather than reporting them to the publicly available system). It is called a shadow because the sensitive information from the environment is kept internal, but the lessons learned (from attack) are available from both inside and the Internet at large. Any tool that is introduced into the test bed would also be required to create a module for this central collection and continue to provide updates for the life of the product, whether it is eventually sold into the federal market or not.

Conclusion

The DREN Cyber Security Test Bed will provide a novel environment for testing new cyber security methods. The following active elements will be implemented as individual, well-contained modules: traditional government off-the-shelf intrusion detection software, traditional commercial off-the-shelf intrusion detection software, active network performance software, and experimental cyber security code. The test bed will be embedded in a production network, thereby providing real traffic to all modules to generate in situ results for comparison, contrast, and correlation. This collocation of cutting-edge security with existing security infrastructure (embedded in a production network) will dramatically expedite the transition of posited network and data protection concepts to proven adaptive cyber security algorithms.

The success of the program relies on not only quality technical implementation but also sound operational and expectation engineering. Creating processes that allow for visibility and interaction from security operations and providing nearly immediate results back to researchers and operations will solidify the value of a given tool and the program as a whole. Bridging the gap between product research and support for the federal environment in the shape of new data feeds, comprehensive aggregate analysis, and response solutions, the goal becomes furthering the overall process, not just the posture of the enterprise security architecture. With so many new attack styles and dimensions, the most valuable outcome of this project will be the new way of approaching the problem. ❑

Mr. Tim Owen received his undergraduate degrees in telecommunications engineering technology from Capitol College in 1989 and physics and science education from North Carolina State University in 1992. He is the chief engineer for WareOnEarth Communications, Inc., and is a key member of the DREN engineering staff. E-mail: towen@hpcmo.hpc.mil

Mr. Rob Scott received his bachelor's degree in communications media from the University of Maryland in 1986. He is currently the chief technology officer of GeoWireless, Inc., and is a key member of the DREN engineering staff. E-mail: rob@hpcmo.hpc.mil

Dr. Roy Campbell received his doctorate in electrical engineering from Mississippi State University in 2002. He currently serves as the program manager for the DREN. E-mail: rcampbell@hpcmo.hpc.mil

References

Baker, W., et al. 2010. 2010 data breach investigations report. http://www.verizonbusiness.com/resources/reports/rp_2010-data-breach-report_en_xg.pdf (accessed October 13, 2010).

Caglayan, A., Toothaker, M., Drapeau, D., Burke, D. and Eaton, G. 2010. Behavioral patterns of fast flux service networks, *Hawaii International Conference on System Sciences (HICSS-43) Cyber Security and Information Intelligence Research Minitrack*. Koloa, Kauai, Hawaii, Jan. 5-8, 2010. http://csdl2.computer.org/comp/proceedings/hicss/2010/3869/00/02-02-05.pdf (accessed October 10, 2010).

Dagon, D. 2005. Botnet detection and response: The internet is the infection. OARC Workshop. http://www.caida.org/workshops/dns-oarc/200507/slides/oarc0507-Dagon.pdf(accessed October 13, 2010).

Delbert, R., et al. April 2010. Shadows in the cloud: Investigating cyber espionage 2.0, JRo3-2010. http://www.nartv.org/mirror/shadows-in-the-cloud.pdf (accessed October 10, 2010).

Economist. July 2010 War in the fifth domain.

Muttik, I. 2010. Coorperation is key to internet security. McAfee Security Journal, 2010 (issue 6): 20–24. http://www.mcafee.com/us/local_content/misc/threat_center/articles/summer2010/msj_article05_cooperation_is_key_to_internet_security.pdf (accessed october 13, 2010).

Rajahalme, J., Conta, A., Carpenter, B., and Deering, S. 2004. IPv6 flow label specification, *Network Working Group Request for Comments, RFC 3697*. Reston, VA: The Internet Society, http://www.ietf.org/rfc/rfc3697.txt (accessed October 13, 2010).

Staniford, S., Moore, D., Paxson, V., and Weaver, N. 2004. The top speed of flash worms, In *Proceedings of the 2004 ACM Workshop on Rapid Malcode*, October 2004. http://www.icir.org/vern/papers/topspeed-worm04.pdf (accessed October 10, 2010).

Symantec. 2010. Symantec Corp. FY2011 2Q Report. http://www.symantec.com/business/theme.jsp?themeid=threatreport (accessed October 13, 2010).

Yan, H., Massey, D., McCracken, E., and Wang L. 2009. BGPMon and NewViews: Real-time BGP. In *Proceedings of the Institute of Electrical and Electronics Engineers International Conference on Computer Communications*, April, Rio de Janeiro, Brazil. http://www.ieee-infocom.org/2009/demos/5%20-%20BGP%20Mon.pdf (accessed October 13, 2010).

Zetter, K. 2010. Google hack attack was ultra-sophisticated, new details show. *WIRED*, January 14, 2010. http://www.wired.com/threatlevel/2010/01/operation-aurora/ (accessed October 13, 2010).